


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

multiple <near/4> parallel <near/4> instance <and> variable



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

multiple near/4 parallel near/4 instance and variable near/3 number near/3 vector

Found 93,756 of 176,279

Sort results by

relevance

Display results

expanded form

[Save results to a Binder](#)
[Search Tips](#)
☐ Open results in a new window

 Try an [Advanced Search](#)

 Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

1 [SVM-based generalized multiple-instance learning via approximate box counting](#)


 Qingping Tao, Stephen Scott, N. V. Vinodchandran, Thomas Takeo Osugi
 July 2004 **Proceedings of the twenty-first international conference on Machine learning ICML '04**

Publisher: ACM Press

 Full text available: pdf(147.99 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

The multiple-instance learning (MIL) model has been very successful in application areas such as drug discovery and content-based image-retrieval. Recently, a generalization of this model and an algorithm for this generalization were introduced, showing significant advantages over the conventional MIL model in certain application areas. Unfortunately, this algorithm is inherently inefficient, preventing scaling to high dimensions. We reformulate this algorithm using a kernel for a support vector ...

2 [Architectures: A perspective on the future of massively parallel computing: fine-grain vs. coarse-grain parallel models comparison & contrast](#)


 Predrag T. Tosić
 April 2004 **Proceedings of the 1st conference on Computing frontiers**

Publisher: ACM Press

 Full text available: pdf(277.49 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

Models, architectures and languages for *parallel computation* have been of utmost research interest in computer science and engineering for several decades. A great variety of parallel computation models has been proposed and studied, and different parallel and distributed architectures designed as some possible ways of harnessing parallelism and improving performance of the general purpose computers. *Massively parallel connectionist models* such as *artificial neural networks* (...

Keywords: cellular automata, distributed systems, massively parallel computing, multiprocessor computers, neural networks, parallel computation models

3 [Level set and PDE methods for computer graphics](#)


 David Breen, Ron Fedkiw, Ken Museth, Stanley Osher, Guillermo Sapiro, Ross Whitaker
 August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

 Full text available: pdf(17.07 MB) Additional Information: [full citation](#), [abstract](#)

Level set methods, an important class of partial differential equation (PDE) methods, define dynamic surfaces implicitly as the level set (iso-surface) of a sampled, evolving nD function. The course begins with preparatory material that introduces the concept of using partial differential equations to solve problems in computer graphics, geometric modeling and computer vision. This will include the structure and behavior of several different types of differential equations, e.g. the level set eq ...

4 Facial modeling and animation



Jörg Haber, Demetri Terzopoulos

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(18.15 MB\)](#) Additional Information: [full citation](#), [abstract](#)

In this course we present an overview of the concepts and current techniques in facial modeling and animation. We introduce this research area by its history and applications. As a necessary prerequisite for facial modeling, data acquisition is discussed in detail. We describe basic concepts of facial animation and present different approaches including parametric models, performance-, physics-, and learning-based methods. State-of-the-art techniques such as muscle-based facial animation, mass-s ...

5 The elements of nature: interactive and realistic techniques



Oliver Deussen, David S. Ebert, Ron Fedkiw, F. Kenton Musgrave, Przemyslaw Prusinkiewicz, Doug Roble, Jos Stam, Jerry Tessendorf

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(17.65 MB\)](#) Additional Information: [full citation](#), [abstract](#)

This updated course on simulating natural phenomena will cover the latest research and production techniques for simulating most of the elements of nature. The presenters will provide movie production, interactive simulation, and research perspectives on the difficult task of photorealistic modeling, rendering, and animation of natural phenomena. The course offers a nice balance of the latest interactive graphics hardware-based simulation techniques and the latest physics-based simulation techni ...

6 Real-time shading



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(7.39 MB\)](#) Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

7 Collision detection and proximity queries



Sunil Hadap, Dave Eberle, Pascal Volino, Ming C. Lin, Stephane Redon, Christer Ericson

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(11.22 MB\)](#) Additional Information: [full citation](#), [abstract](#)

This course will primarily cover widely accepted and proved methodologies in collision

detection. In addition more advanced or recent topics such as continuous collision detection, ADFs, and using graphics hardware will be introduced. When appropriate the methods discussed will be tied to familiar applications such as rigid body and cloth simulation, and will be compared. The course is a good overview for those developing applications in physically based modeling, VR, haptics, and robotics.

8 GPGPU: general purpose computation on graphics hardware



David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff Woolley, Aaron Lefohn

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: [pdf\(63.03 MB\)](#) Additional Information: [full citation](#), [abstract](#)

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

9 Parallel execution of prolog programs: a survey



Gopal Gupta, Enrico Pontelli, Khayri A.M. Ali, Mats Carlsson, Manuel V. Hermenegildo

July 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 23 Issue 4

Publisher: ACM Press

Full text available: [pdf\(1.95 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Since the early days of logic programming, researchers in the field realized the potential for exploitation of parallelism present in the execution of logic programs. Their high-level nature, the presence of nondeterminism, and their referential transparency, among other characteristics, make logic programs interesting candidates for obtaining speedups through parallel execution. At the same time, the fact that the typical applications of logic programming frequently involve irregular computatio ...

Keywords: Automatic parallelization, constraint programming, logic programming, parallelism, prolog

10 Compiling nested data-parallel programs for shared-memory multiprocessors



Siddhartha Chatterjee

July 1993 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 15 Issue 3

Publisher: ACM Press

Full text available: [pdf\(4.17 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)

Keywords: compilers, data parallelism, shared-memory multiprocessors

11 Parallel instance discrete-event simulation using a vector uniprocessor

James F. Ohi, Bruno R. Preiss

December 1991 **Proceedings of the 23rd conference on Winter simulation**

Publisher: IEEE Computer Society

Full text available: [pdf\(730.43 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

Keywords: SIMD processors, parallel simulation, stochastic modelling, vector processors

12 Crowd and group animation



Daniel Thalmann, Christophe Hery, Seth Lippman, Hiromi Ono, Stephen Regelous, Douglas Sutton

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: pdf(20.19 MB) Additional Information: [full citation](#), [abstract](#)

A continuous challenge for special effects in movies is the production of realistic virtual crowds, in terms of rendering and behavior. This course will present state-of-the-art techniques and methods. The course will explain in details the different approaches to create virtual crowds: particle systems with flocking techniques using attraction and repulsion forces, copy and pasting techniques, agent-based methods. The architecture of software tools will be presented including the MASSIVE softwa ...

13 WSQ/DSQ: a practical approach for combined querying of databases and the Web



Roy Goldman, Jennifer Widom

May 2000 **ACM SIGMOD Record , Proceedings of the 2000 ACM SIGMOD international conference on Management of data SIGMOD '00**, Volume 29 Issue 2

Publisher: ACM Press

Full text available: pdf(223.65 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We present WSQ/DSQ (pronounced "wisk-disk"), a new approach for combining the query facilities of traditional databases with existing search engines on the Web. WSQ, for *Web-Supported (Database) Queries*, leverages results from Web searches to enhance SQL queries over a relational database. DSQ, for *Database-Supported (Web) Queries*, uses information stored in the database to enhance and explain Web searches. This paper focuses primarily on WSQ, describing a simple, lo ...

14 Architectural and application: the performance of the NEC SX-4 on the NCAR benchmark suite

Steven W. Hammond, Richard D. Loft, Philip D. Tannenbaum

November 1996 **Proceedings of the 1996 ACM/IEEE conference on Supercomputing (CDROM) Supercomputing '96**

Publisher: IEEE Computer Society

Full text available: pdf(137.27 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

In November 1994, the NEC Corporation announced the SX-4 supercomputer. It is the third in the SX series of supercomputers and is upward compatible from the SX-3R vector processor with enhancements for scalar processing, short vector processing, and parallel processing. In this paper we describe the architecture of the SX-4 which has an 8.0 ns clock cycle and a peak performance of 2 Gflops per processor. We also describe the composition of the NCAR Benchmark Suite, designed to evaluate the ...

15 Compiler transformations for high-performance computing



David F. Bacon, Susan L. Graham, Oliver J. Sharp

December 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 4

Publisher: ACM Press

Full text available: pdf(6.32 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

In the last three decades a large number of compiler transformations for optimizing programs have been implemented. Most optimizations for uniprocessors reduce the number of instructions executed by the program using transformations based on the

analysis of scalar quantities and data-flow techniques. In contrast, optimizations for high-performance superscalar, vector, and parallel processors maximize parallelism and memory locality with transformations that rely on tracking the properties o ...

Keywords: compilation, dependence analysis, locality, multiprocessors, optimization, parallelism, superscalar processors, vectorization

16 Survey and taxonomy of packet classification techniques



David E. Taylor

September 2005 **ACM Computing Surveys (CSUR)**, Volume 37 Issue 3

Publisher: ACM Press

Full text available: pdf(1.60 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Packet classification is an enabling function for a variety of Internet applications including quality of service, security, monitoring, and multimedia communications. In order to classify a packet as belonging to a particular flow or set of flows, network nodes must perform a search over a set of filters using multiple fields of the packet as the search key. In general, there have been two major threads of research addressing packet classification, algorithmic and architectural. A few pioneerin ...

Keywords: Packet classification, flow identification

17 Machine Learning Methods for Predicting Failures in Hard Drives: A Multiple-Instance Application

Joseph F. Murray, Gordon F. Hughes, Kenneth Kreutz-Delgado

September 2005 **The Journal of Machine Learning Research**, Volume 6

Publisher: MIT Press

Full text available: pdf(274.51 KB) Additional Information: [full citation](#), [abstract](#)

We compare machine learning methods applied to a difficult real-world problem: predicting computer hard-drive failure using attributes monitored internally by individual drives. The problem is one of detecting rare events in a time series of noisy and nonparametrically-distributed data. We develop a new algorithm based on the multiple-instance learning framework and the naive Bayesian classifier (mi-NB) which is specifically designed for the low false-alarm case, and is shown to have promising p ...

18 Shape-based retrieval and analysis of 3D models



Thomas Funkhouser, Michael Kazhdan

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: pdf(12.56 MB) Additional Information: [full citation](#), [abstract](#)

Large repositories of 3D data are rapidly becoming available in several fields, including mechanical CAD, molecular biology, and computer graphics. As the number of 3D models grows, there is an increasing need for computer algorithms to help people find the interesting ones and discover relationships between them. Unfortunately, traditional text-based search techniques are not always effective for 3D models, especially when queries are geometric in nature (e.g., find me objects that fit into thi ...

19 Parallel programming with control abstraction



Lawrence A. Crowl, Thomas J. LeBlanc

May 1994 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 16 Issue 3

Publisher: ACM Press

Full text available: pdf(3.68 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#),

[review](#)

Parallel programming involves finding the potential parallelism in an application and mapping it to the architecture at hand. Since a typical application has more potential parallelism than any single architecture can exploit effectively, programmers usually limit their focus to the parallelism that the available control constructs express easily and that the given architecture exploits efficiently. This approach produces programs that exhibit much less parallelism that exists in the applic ...

Keywords: architectural adaptability, closures, control abstraction, data abstraction, early reply, multiprocessors, parallel programming languages, performance tuning

20 [Automatic translation of FORTRAN programs to vector form](#)



Randy Allen, Ken Kennedy

October 1987 **ACM Transactions on Programming Languages and Systems (TOPLAS)**,

Volume 9 Issue 4

Publisher: ACM Press

Full text available: [pdf\(3.14 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The recent success of vector computers such as the Cray-1 and array processors such as those manufactured by Floating Point Systems has increased interest in making vector operations available to the FORTRAN programmer. The FORTRAN standards committee is currently considering a successor to FORTRAN 77, usually called FORTRAN 8x, that will permit the programmer to explicitly specify vector and array operations. Although FORTRAN 8x will make it convenient to specify explicit vector ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)